MAINCRAFTS
TECHNOLOGY

House Price Prediction using Linear Regression

Task 1 – Artificial Intelligence & Machine Learning

1. Introduction

The objective of this project is to build and evaluate a **Linear Regression model** to predict house prices using the **California Housing dataset**.

This task introduces the complete **machine learning workflow**, including data loading, exploratory data analysis (EDA), preprocessing, model training, evaluation, and reporting.

Linear Regression is used as a baseline model to understand the relationship between housing features and median house values.

2. Dataset Description

The **California Housing dataset** contains information collected from the 1990 California census. Each row represents a housing district with the following features:

- Median income
- House age
- Average number of rooms
- Average number of bedrooms
- Population
- Average occupancy
- Latitude
- Longitude

Target Variable:

- Median house value

The dataset was sourced from **Kaggle**, which is an acceptable alternative to the built-in scikit-learn dataset.

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed to understand the structure and distribution of the data.

Key EDA Steps:

- Checked dataset shape and feature names
- Verified missing values
- Analyzed feature distributions using plots
- Examined correlations between features and the target variable

Observations:

- Median income shows a strong positive correlation with house prices
- Some features contain outliers that may affect model performance
- Feature scales vary significantly, making scaling necessary

EDA helped guide preprocessing and model selection.

4. Data Preprocessing

Before training the model, the following preprocessing steps were applied:

- **Feature–target split** (X and y)
- **Train-test split** to evaluate generalization
- **Feature scaling** using standardization to ensure all features contribute equally

Preprocessing ensures that the Linear Regression model performs optimally and avoids bias due to feature magnitude differences.

5. Model Training

A **Linear Regression model** from the scikit-learn library was used.

Steps:

- Model initialization
- Training on the scaled training dataset
- Prediction on test data

Linear Regression was chosen because it is:

- Simple and interpretable
- A strong baseline for regression problems
- Useful for understanding feature impact

6. Model Evaluation

The model was evaluated using standard regression metrics:

- **Mean Absolute Error (MAE)**
- **Root Mean Squared Error (RMSE)**
- **R² Score**

Results:

- MAE indicates the average prediction error
- RMSE penalizes larger errors more heavily
- R² score explains how well the model fits the data

These metrics show that the model provides a reasonable baseline performance but has room for improvement.

7. Model Saving

The trained model was saved as a **pickle file (.pkl)**, allowing reuse without retraining. This is useful for deployment or integration into future applications.

8. Future Improvements

Although the Linear Regression model performs adequately, several improvements can be explored:

- **Feature Engineering:** Create new features or apply transformations to capture complex relationships
- **Polynomial Regression:** Model non-linear patterns in house prices
- **Regularization:** Use Ridge or Lasso regression to reduce overfitting
- **Outlier Handling:** Identify and treat extreme values
- **Advanced Models:** Compare performance with models like Random Forest or Gradient Boosting

These enhancements could significantly improve prediction accuracy.

9. Conclusion

This project successfully demonstrates the **end-to-end machine learning workflow** using Linear Regression.

From data exploration to model evaluation and saving, all essential steps were implemented according to task guidelines.

The project serves as a strong foundation for more advanced regression techniques and is suitable for portfolio presentation.